PSM Insight: The Army-DoD Tool to Implement Issue-Driven Software Measurement

Don Scott Lucero
U.S. Army Software Metrics Office

The Army Software Metrics Office has developed a software measurement tool, PSM Insight, to implement the Army's issue-driven software measurement policy [1] and the practical software measurement (PSM) process [2]. The PSM Insight tool supports tailoring and controlling the data needed to implement an effective measurement program. This article describes some of the challenges faced by the PSM Insight Development Team to deliver a tool with the flexibility to support a tailored measurement process.

The PSM Process

PSM is a Department of Defense (DoD)-sponsored project to provide program managers with the objective information needed to successfully manage software-intensive projects. PSM is based on software measurement experience with DoD and industry projects. Measurement professionals from DoD, industry, and academia have collaborated to define best measurement practices used within the software acquisition and engineering communities.

PSM treats measurement as a flexible process, not a predefined list of graphs or reports. The process is adapted to address the specific software issues, objectives, and information requirements unique to each project. The PSM process is defined by a set of nine best practices, called measurement principles. The underlying objective of the PSM guidelines is to integrate the measurement requirements into the software process. Software measures are tailored to reflect the existing project management and software development processes, ensuring that the measures provide meaningful and cost-effective results. The measurement process also is integrated with existing risk and financial management processes to provide a basis for objective decision making.

The PSM process is defined in the PSM Guide, *Practical Software Measurement: A Foundation for Objective Project Management*. The guide explains the basic concepts of the software measurement process, offers detailed implementation guidance, and provides realistic case studies of software measurement used on typical projects. In addition to the guide, the PSM project provides training and workshops on the PSM process, hands-on management support as requested by other projects, and the PSM Insight tool to manage data.

Development of PSM Insight started in February 1997 with a Memorandum of Agreement between the Army Software Metrics Office and the PSM Steering Group. This agreement was to develop a single tool to support an integrated DoD software measurement strategy. PSM Insight provides a PC-based management capability to implement the PSM process. PSM Insight guides managers through tailoring software measures to their specific project and provides the data management capabilities to select and monitor project-specific indicators, measures, and data items.

The PSM Guide presents a systematic but flexible, issue-driven measurement process with examples of data items, issues, categories, and measures. PSM Insight supports the PSM process by importing, storing, and graphing these minimum examples, but also provides a flexible database that allows a user to define project-specific data items, issues, categories, and measures. The PSM Insight tool provides desktop support for a software measurement project that achieves the nine fundamental principles of the PSM process, as illustrated in Table 1.

The PSM Insight Tool

PSM Insight is a Windows-based application that allows a high level of flexibility in data management including data modification, data browsing, and sophisticated graphing capabilities. PSM Insight has the capability to tailor software measures to unique project issues and allows using data already available from an existing software development process. The tool has been designed in compliance with industry standards for Open Database Connectivity, which allows the tool to create or automatically access different databases with dissimilar data formats.

PSM Insight can accept any unique data parameter required for a software measurement project and can manage data according to the attributes and software components,

Table 1. *The nine fundamental principles of the PSM process.*

- Project issues and objectives drive the measurement requirements.
- The developer's software process defines how the software is measured.
- Collect and analyze data at a level of detail sufficient to identify and isolate software problems.
- Implement an independent analysis capability.
- Use a systematic analysis process to trace the measures to the decisions.
- Interpret the measurement results in the context of other project information.
- Integrate software measurement into the project management process throughout the lifecycle.
- Use the measurement process as a basis for objective communications.
- Focus initially on project-level analysis.

organizations, or activities.

Data Items, Attributes, and Structures

A data item is a specific type of information collected to manage and monitor a project. Data items quantify what is being measured. Common data items include "Start Date," "End Date," "Number of Test Cases," "Cost," "Number of Requirements," and "Defects."

Attributes are characteristics or properties of a measure that distinguish one data item from another. Common attributes include "Planned/Actual," "Version," or "Language."

The structure defines the organization of a project and identifies the level at which data items are collected. For example, this could be a "Software Process Activity" structure of work task that must be completed in the project, a "Software Component" structure of software products that make up the entire project, or "Lines of Code" collected at the organizational level or by functional unit.

Because of the size limitations of this article, only a few examples of the PSM Insight tool displays are given in Figures 1 through 3. Figure 1 shows the Main Menu Window of PSM Insight, which allows the user to create project-specific software measures. This display provides the user with three icon groups to define the tool's top-level functions: Select the Project, Tailor the Measures, and Apply the Measures.

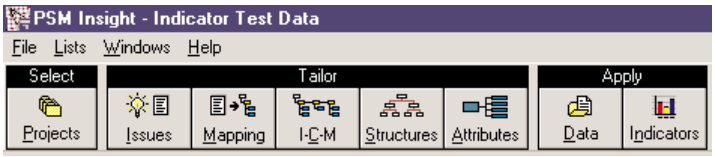


Figure 1. *Main Menu Window of PSM Insight.*

The Projects icon, in the Main Menu Window, allows the PSM Insight user to define or select a project on which software measures are collected.

The Tailor icon group allows the user to define and tailor the software measurement project. It includes five functional icons: Issues, Mapping, I-C-M (Issues-Categories-Measures), Structures, and Attributes.

The Project-Specific Issue List, shown in Figure 2, allows the user to list all issues that are critical to the success of the project.

The Indicators function allows the user to view graphic displays of the measurement data, shown in Figure 3.

For a full description of PSM Insight tool displays, visit the PSM Web site at www.psmsc.com.

The Challenges of Building PSM Insight

The Open Requirements-Definition Process

PSM Insight is being developed in a rapid-prototyping process using Borland's Delphi, Version 2.0. This environment allows a nonproprietary, run-time program to be created for Windows 3.1, Windows 95, and Windows NT. The Windows-based interface to the advanced Delphi design capabilities provides a powerful tool for information analysis and retrieval.

The initial requirements for PSM Insight were defined from the Army's experience in applying previous metrics management tools, including Metrics Guided Maturity (MGM) and Software Metrics Management Information System (SMMIS). PSM Insight requirements initially were defined through internal meetings of the PSM Insight Development Team. The process to define user requirements greatly expanded after the first year of development, when the first prototype was available for demonstration. PSM Insight became the subject of a requirements-definition workshop at the first PSM User's Group Conference. The workshop allowed participants to evaluate the existing PSM Insight requirements and recommend new attributes for future versions. The challenge the development team faced was how to reach a consensus on the requirements baseline, while allowing any interested party to suggest new features or enhancements to the tool. However, the contributions from these open workshops have proven invaluable for focusing attention on common areas of concern and for prioritizing future design work.

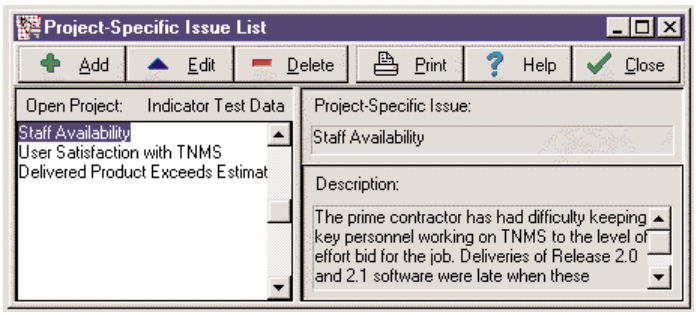
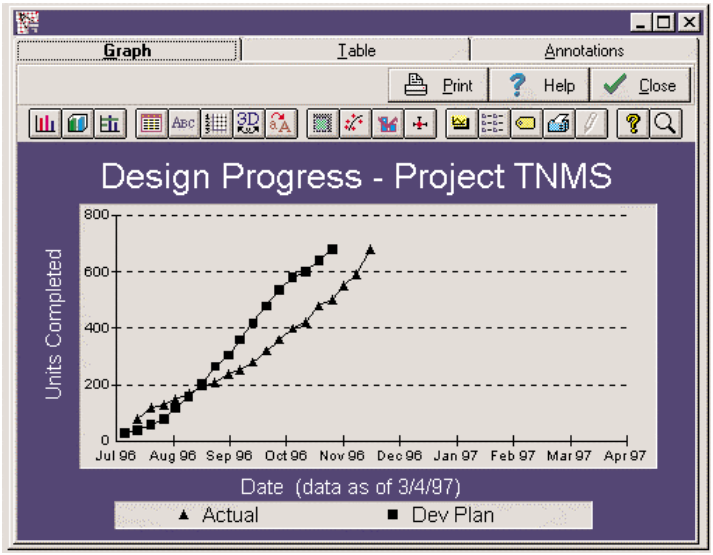


Figure 2. *Project-Specific Issue List.*

The challenge in the open requirements-definition process is how to effectively and diplomatically handle requests that conflict or interfere with each other. For example, a common conflict in user expectations is the level and complexity of the security features of PSM Insight. User requests range from completely open access to all PSM Insight features and data, to restriction of access based on multilevel roles in an organization. This open requirements-definition process has also created more work for the PSM Insight Development Team. The team had to tabulate, evaluate, and assign priorities for each new tool

Figure 3. *Indicators function.*



requirement. The team implemented the highest-priority requirements first, while incrementally building essential capabilities required in the objective system.

Supporting Changes to the PSM Guide

PSM Insight's formal requirements base, the PSM Guide, is reissued about once a year. Terminology, definitions, and emphasis in the guide are modified to address changing needs and lessons learned in the measurement process. An important feature of PSM Insight is to provide the approved PSM Guide's tree of successive I-C-Ms as templates during the tailoring process. A user may select, rename, edit, remove, or add to the I-C-M templates that are suggested by PSM Insight at each step. These items are embedded in PSM Insight as text descriptions, preset value lists, and online help. The recommended I-C-M templates are further described through training examples and sample data sets in the tool. Because of the extensive use of embedded descriptions, the PSM Insight Development Team is faced with a major challenge each time the I-C-M templates are changed. During the development effort, changes in the PSM Guide have caused entire issues, categories, measures, and measurement elements to be renamed, removed, or redefined.

Another PSM Insight design challenge has been to allow the tool to be tailored to each user, such as providing the capability to create and modify project-specific I-C-M templates. By allowing the tool to be tailored, the challenge has been to provide the ability to define and refine the way an evolving project is organized. The development team is trying to determine how to build measures that span multiple projects. Tailoring a measurement program tool to organizational structures requires the capability to combine information across multiple projects that may be configured differently, with data collected at different levels and reported at various frequencies.

Meeting the Users' Technical Capabilities

A major development challenge of PSM Insight has been to build a tool that will be useful for both the novice user of measurement data and the experienced manager. The PSM Guide directs a user through a process of tailoring software measures to specific project issues. This process starts with a defined set of six "common issues" that apply to any project. The PSM Guide then assigns each common issue a set of "issue categories" that are linked to appropriate measures. A primary requirement of the tool is to lead users through the same scheme of I-C-Ms.

However, this process is a nuisance to experienced managers who already know the measures and data they need to collect. The PSM Insight graphical user interface requires the user to walk through the I-C-Ms scheme that is prescribed in the PSM Guide. This may result in an additional step for the experienced manager and therefore violates a PSM Insight design objective that additional effort in managing data must be minimized. A related design objective is to minimize entering and storing repetitive, placeholder data.

The question remains whether the conceptual model is adequate to reflect real-life measuring and reporting needs. PSM Insight uses structures, attributes, and data items to define a measure. Thus far, these data types have been sufficient to define the measures in the guide and user-defined measures. As the PSM Guide moves from project-level measures to include enterprise- or organization-level measures, the PSM Insight Development Team will need to assess if additional constructs are needed.

Creating Flexibility

PSM Insight needed to retain simplicity and ease of use, while offering the flexibility required for a tailored measurement process. More flexibility frequently requires more options to be made available to the tool users, rather than fixed by the tool designers. However, the number of decisions may become overwhelming. In response, PSM Insight makes extensive use of default values to minimize the administrative burden of answering every possible tailoring option.

Another design flexibility issue is allowing a user to select a specific display view for the measurement data. For example, any data can be presented as either an interval measurement (number of problems opened each month) or a cumulative number (all problems opened to date). To assist in a tailored measurement process, PSM Insight can translate between the two representations when a user selects a specific graph. Measurements that are stored in PSM Insight as "quantity-per-time-interval" can be viewed as "total-to-date" when needed. Using stored data in arithmetic calculations is a complex design problem, especially when aggregating lower-level data upward.

Training Users

Every software application must achieve a balance between advanced, complex capabilities and the ability of a user to understand and work the program. The data-handling flexibility and capabilities for local user customization have increased the complexity of PSM Insight. Subsequently, the PSM Insight Development Team had to design resident features to support usability of the tool. Two methods were implemented: extensive help screens and portable training.

PSM Insight can be a challenging package to learn. Most users take a one- or three-day class on the PSM method to acquire a working knowledge of the terminology and principles behind the software measurement process. On-site workshops are the best way to learn PSM Insight. Expert instructors teach the basics of PSM Insight, provide help in applying PSM to local projects and tailor the PSM Insight software to site-specific needs. For large sites, this is the optimum learning experience. For smaller sites and one-person operations, the PSM Insight Development Team is building a computer-based tutorial (CBT). The multimedia tutorial will review terminology, illustrate the basic features of PSM Insight, present a demonstration of each major function, and provide guided practice sessions for interactive learning opportunities. The CBT simulates PSM Insight as it walks the student through the major features of the tool, providing tips along the way and prompting the student for input during practice sessions. Although a tutorial cannot address site-specific questions, it can help the self-motivated user become familiar enough with the software to explore more advanced features independently.

Using Commercial-Off-the-Shelf (COTS) Products

The Rapid Application Development environment of Delphi allows the PSM Insight implementers to use tools that are more intuitive and visual. In addition, the tools and components can

be extended to include third-party components, "widgets," and COTS products.

PSM Insight development has increased code reusability and programming productivity with the use of third-party components and COTS products. Many of PSM Insight's important features have been implemented using these products. However, COTS products introduce their own set of problems and challenges. Listed below are some of the COTS products and third-party components now in the tool, followed by the technical challenges and problems faced when implementing the components.

COTS products and third-party components

- Graphing of Indicators is implemented using Graphics Server from Pinnacle Publishing. Graphics Server is a comprehensive graphing toolkit consisting of a core set of graphing routines that PSM Insight accesses through library functions. Several hundred graphing parameters can be passed to Graphics Server to control displaying, storing, and printing of graphs.
- The Tailoring Reports capability has been implemented with the help of Seagate Crystal Reports. Crystal Reports is a powerful Windows reporting tool that helps PSM Insight dynamically tailor reports to the user's needs. The PSM Insight Delphi code accesses Crystal Reports' dynamic link library (DLL) for sophisticated report generation and printing capabilities.
- The outline lists used in the tool were implemented with a native Delphi component from a third-party developer. This compiled unit gave programmers access to many properties and events to control the display and action of the outline lists.
- The search data and sort data tasks were implemented with third-party components, saving many hours of programming time.

Technical Problems — distributing and installing the run-time DLLs associated with the COTS products:

- Different installations routines had to be provided for 16- and 32-bit environments.
- Identifying when to overwrite existing DLLs had to be defined, since other applications may depend on them.

Technical Problems — saving, storing, and recalling parameters associated with each COTS product:

- PSM Insight must be able to save, retain, and pass data to the COTS products for a seamless integration of the products for the end user.
- Since each DLL is a special type of executable file or application, it has its own set of bugs that have to be handled or avoided. Therefore, code was written to tell PSM Insight what to do when a call to a function fails.
- Vendor support for bug fixes and product upgrades caused some schedule delays.

Conclusion

PSM Insight supports the PSM issue-driven measurement process by providing examples presented in the PSM Guide and allowing user-defined issues. Because PSM Insight's design is compliant with industry standards for Open Database Connectivity, PSM Insight can accept data from an existing software development process.

The Army's experience with previous metrics management tools and the use of COTS products expedited the development of PSM Insight.

PSM Insight can be obtained at no cost through the Army Software Metrics Office. To obtain a copy of the tool, or for more information on the issue-driven software measurement process, visit the Army Software Metrics Office Web site at www.ArmySoftwareMetrics.org or the PSM Web site at www.psmsc.com.

Note: The PSM process has served as the basis for ISO/IEC 15939, Software Measurement Process Framework. The PSM process may change with the creation of ISO/IEC 15939 to support the standards community. ISO/IEC 15939 may be more formally implemented in PSM Insight. Readers can access www.iso14001.net/iso15939/ to read the "Scope of the proposed ISO/IEC 15939 standard." ◆

Acknowledgments

I acknowledge Dave Morris of IEI for his efforts in developing PSM Insight. Jack McGarry managed the PSM project for years, and Cheryl Jones currently manages the project. More information may be obtained through the PSM Support Center:

U.S. Army
TACOM-ARDEC
AMSTA-AR-QAT-A, Building 62
Picatinny Arsenal, NJ 07806-5000
Voice: 973-724-5638
Fax: 973-724-2382
E-mail: psm@pica.army.mil
Internet: www.psmsc.com

About the Author

Don Scott Lucero is a software engineer on the headquarters staff of the Army's Operational Test and Evaluation Command (OPTEC). He is responsible for the Army's Software Metrics Office as well as OPTEC's software test and evaluation policy and methods. He has 15 years experience working on Army software development projects and has a bachelor's and a master's degree in computer science.

U.S. Army Software Metrics Office
Attn: CSTE-OM-SMO, Park Center IV
4501 Ford Avenue
Alexandria, VA 22302-1458
Voice: 703-681-3895
Fax: 703-681-6914
E-mail: lucerodon@hq.optec.army.mil

References

1. Office, Director of Information Systems for Command, Control, Communications, and Computers (DISC4, SAIS-ADW) Memorandum, "Acquisition Reform and Software Metrics," dated Sept. 19, 1996.
2. *Practical Software Measurement (PSM): A Foundation for Objective Project Management*, Version 3.1a, April 17, 1998.